WHAT IS CLAIMED IS:

1. An arithmetic logic unit over a finite field GF (2^m), comprising:

a control unit for generating control signals required for an RS-block unit, an SR-block unit and a UV-block unit while outputting an externally-applied signal (mult/div) to be used as an input to select multiplication and division operations without change;

the RS-block unit for generating an output value (r_0) when receiving the control signals from the control logic unit, and transmitting the output value (r_0) to the control logic unit, and calculating R and S values of multiplication and division algorithms;

the SR-block unit for performing multiplication and division operations when receiving a control signal output from the control logic unit and a value state output from a one-bit register (state) of the control logic unit, and shifting register values in right and left directions; and

a UV-block unit for outputting one-bit register values (P_{m-1}/u_0 and a_0/v_0) to the control logic unit when receiving the control signals from the control logic unit, and calculating U and V values of multiplication and division algorithms.

2. The arithmetic logic unit according to claim 1, wherein the control logic unit comprises:

an AND gate (G1) for receiving the output value (state) from the one-bit register (state), and also receiving an output value (b_i/z-flag) from the SR-block unit through an inverter;

an AND gate (G2) for receiving the output value (r_0) from the RS-block unit, and also receiving the output value (state) from the one-bit register (state) through an inverter;

an AND gate (G3) for receiving the output value (state) from the one-bit register (state), and updating a value output from a register (c-flag) when receiving the output value (b_i/z-flag) from the SR-block unit;

an AND gate (G4) for receiving the output value (r_0) from the RS-block unit, and also receiving the output value (a_0/v_o) from the UV-block unit;

an AND gate (G5) for receiving the output value (r_0) from the RS-block unit, and outputting a control signal (Ctrl3) to the RS-block unit when receiving the output value (state) from the one-bit register;

an OR gate for outputting a signal used to update the output value (state) from the one-bit register (state) using values output from the AND gates (G1 and G2);

an XOR gate for outputting a control signal (Ctrl2) to the UV-block unit using a value output from the AND gate (G4) and the value (P_{m-1}/u_0) output from the UV-block unit; and

the register (c-flag) for outputting the value (c-flag) to the SR-block unit using a value output from the AND gate (G3).

3. The arithmetic logic unit according to claim 2, wherein the register (c-flag) is initialized to "1" when starting a division while operating together with the SR-block unit.

4. The arithmetic logic unit according to claim 1, wherein the RS-block unit is constructed so that structures are arranged in cascade, the structures each comprising:

a multiplexer for receiving values output from registers (r and s), and a control signal (Ctrl3) output from the control logic unit;

an AND gate for receiving a value output from the register (s) and a control signal (Ctrl1) output from the control logic unit;

an XOR gate for receiving a value output from the register (r) and a value output from the AND gate;

the register (r) for generating the register value (r_0) using a value output from the XOR gate and outputting the register value (r_0) to the control logic unit; and

the register (s) for generating a register value (s) using a value output from the multiplexer.

5. The arithmetic logic unit according to claim 1, wherein the SR-block unit is constructed so that structures are arranged in cascade, the structures each comprising:

an OR gate for receiving the signal (mult/div) from the control logic unit through an inverter, and also receiving the output value (state) from the one-bit register (state) of the control logic unit;

a multiplexer for outputting a signal used to update a register (b_i/z-flag) using a value output from the OR gate, the value (c-flag) output from the register (c-flag) of the control logic unit and a value fed back from the register (b_i/z-flag); and

a register (b_i/z-flag) updated using the update signal output from the multiplexer, and then feeding back the updated signal to the multiplexer, while providing the value (b_i/z-flag) output from the register (b_i/z-flag) to the control logic unit.

6. The arithmetic logic unit according to claim 5, wherein the register (b_i/z-flag) is implemented by an m-bit bidirectional shift register, instead of a $\log_2(m+1)$-bit counter, so as to operate a count value of a division algorithm.

7. The arithmetic logic unit according to claim 1, wherein the UV-block unit comprises:

structures which are arranged in cascade, the structures each having a register for outputting the register value (P_{m-1}/u_0) to the control logic unit, a register for outputting the register value (a_0/v_0) to the AND gate of the control logic unit, and a multiplexer, AND gates and XOR gates for updating the values output from the registers;

an AND gate for consistently generating "0(zero)" in a multiplication mode so as to allow the multiplexer to select the value (a_0/v_0) output from the register (a_0/v_0) in response to the signals (mult/div and Ctrl3) output from the control logic unit; and

an AND gate for consistently generating "0(zero)" in a division mode using the signal (mult/div) output from the control logic unit and the value (a_0/v_0) output from the register (a_0/v_0).

8. The arithmetic logic unit according to claim 1, wherein the division algorithm is implemented based on a binary greatest common divisor algorithm.

9. A Galois Field arithmetic logic unit for performing arithmetic operations comprising a multiplication operation and a division operation over the Galois field GF(2^m), the arithmetic logic unit comprising:

a control logic unit configured to receive an operation-type indication regarding whether to perform the multiplication operation or the division operation, the control logic unit comprising at least one control logic latch, and a plurality of control logic gates to receive control inputs and to provide control outputs for performing the arithmetic operations using input and output registers sharing common logic for both multiplication and division operations;

a first logic block including a first and second set of first-logic-block latches and associated first-logic-block gates, the first logic block operable to transfer data contents from the second set of first-logic-block latches to the first set of first-logic-block latches, the first logic block further operable to compute a first-logic-block exclusive OR value on the contents of the first and second set of first-logic-block latches;

a second logic block including a set of second-logic-block latches and associated second-logic-block gates, the second logic block including a multiplexer and at least one second-logic-block gate arranged to configurably permit the second logic block to operate as a bidirectional shift register; and

a third logic block including a first and second set of third-logic-block latches and associated third-logic-block gates, the third-logic-block gates configured to selectably produce a third-logic-block exclusive OR operation on the contents the first and second set of third-logic-block latches or on an externally provided set of bits.

10. The Galois Field arithmetic logic unit according to claim 9, wherein the control logic unit comprises:

a one-bit state latch having an associated state-latch value that is determined by a state latch function of the sum of the product of the state-latch value and the inverse of a second-logic-block output control signal and the product of the inverse of the state-latch value and a first-logic-block output control signal; and

a one-bit c-flag latch having an associated c-flag value that is determined by the product of the second-logic-block output control signal and the state latch value.

11. A method for performing multiplication and division operations over the finite field GF(2^m), the method comprising:

receiving an operation-type indication regarding whether to perform the multiplication operation or the division operation, at a control logic unit comprising at least one control logic latch;

providing control signals, based on the operation-type indication, to at least one logic block that contains shared logic gates for performing the multiplication and division operations, the shared logic gates forming a logic structure including a configurable shared bidirectional shift register and at least one exclusive OR operator;

maintaining a state value based on a combination of the present state and a control signal from the logic block; and

computing an output value consistent with the multiplication and division operations based on the state value, input values, and the operation-type indication.